



PATENT ABSTRACTS OF JAPAN

(11) Publication number: **09179911 A**(43) Date of publication of application: **11 . 07 . 97**

(51) Int. Cl.

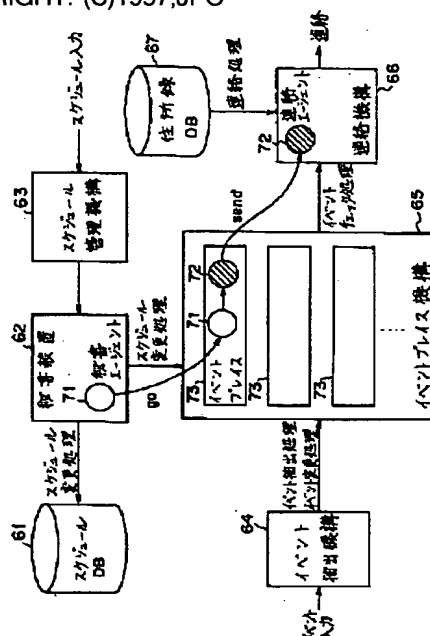
G06F 17/60**G06F 15/02**(21) Application number: **07339686**(71) Applicant: **FUJITSU LTD**(22) Date of filing: **26 . 12 . 95**(72) Inventor: **NIHORI YASUHIRO****(54) SYSTEM AND METHOD FOR ELECTRONIC SECRETARY**

COPYRIGHT: (C)1997,JPO

(57) Abstract:

PROBLEM TO BE SOLVED: To notice a user of trouble information that hinders the schedule of a user before appointed time.

SOLUTION: Schedule information which is inputted to a schedule managing mechanism 63 is stored in a schedule data base 61 by a secretary device 62. At this time, the secretary device 62 generates a secretary agent 71 which monitors a specific line of a means of transport and sends it to an event place mechanism 65. When event information inputted from an event extracting mechanism 64 relates to the line to be monitored, the secret agent 71 sends a communication agent 72 to the communication mechanism 66. The communication mechanism 66 refers to an address book data base and notice the user of the event information by a specified communication method. Consequently, the user can speedily obtain trouble information on the used means of transport at his or her visiting place.



(2) - 2

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-179911

(43) 公開日 平成9年(1997)7月11日

(51) Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 17/60			G 0 6 F 15/21	L
15/02	3 5 5		15/02	3 5 5 A

審査請求 未請求 請求項の数13 O L (全 17 頁)

(21) 出願番号 特願平7-339686

(22) 出願日 平成7年(1995)12月26日

(71) 出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中4丁目1番
1号

(72) 発明者 新堀 恭裕

神奈川県川崎市中原区上小田中1015番地
富士通株式会社内

(74) 代理人 弁理士 大曾 義之 (外1名)

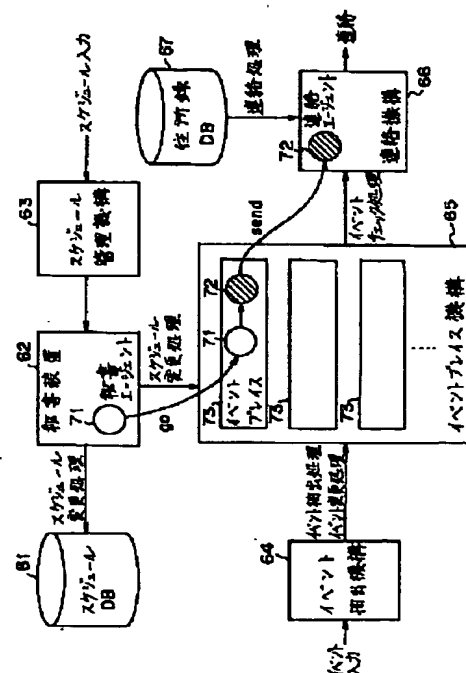
(54) 【発明の名称】 電子秘書システムおよび方法

(57) 【要約】

【課題】 ユーザのスケジュールを妨げる障害情報を、予定時刻以前にユーザに通知することが課題である。

【解決手段】 スケジュール管理機構63に入力されたスケジュール情報は、秘書装置62によりスケジュールデータベース61に格納される。このとき、秘書装置62は交通機関の特定の路線を監視する秘書エージェント71を生成し、イベントプレイス機構65に送り込む。イベント抽出機構64から入力されたイベント情報が監視対象の路線に関係する場合は、秘書エージェント71は、連絡エージェント72を連絡機構66に送る。連絡機構66は住所録データベースを参照して、指定された連絡方法でイベント情報をユーザに通知する。これにより、ユーザは、利用する交通機関の障害情報を外出先でも迅速に得ることができる。

電子秘書システムの構成図



【特許請求の範囲】

【請求項1】 スケジュール管理を行う情報処理システムにおいて、
ユーザのスケジュール情報を管理する管理手段と、
前記ユーザの利用する資源に関するイベント情報の入力
を検出する検出手段と、
前記スケジュール情報と前記検出手段により検出された
前記イベント情報とに基づいて、前記スケジュール情報
に含まれる資源に関するイベント情報を前記ユーザに通
知する通知手段と、
を備えることを特徴とする電子秘書システム。

【請求項2】 前記検出手段は、前記ユーザが利用する
交通機関に関する障害情報を、前記ユーザの利用する資
源に関するイベント情報として検出し、前記通知手段
は、前記スケジュール情報に含まれる交通機関に関する
障害情報を前記ユーザに通知することを特徴とする請求
項1記載の電子秘書システム。

【請求項3】 前記管理手段は、前記スケジュール情報
を資源毎に管理し、前記通知手段は、前記検出されたイ
ベント情報を資源毎に該スケジュール情報と比較して、
該検出されたイベント情報が前記スケジュール情報に含
まれる資源に関係するかどうかを判定することを特徴と
する請求項1記載の電子秘書システム。

【請求項4】 前記通知手段は、前記検出されたイベン
ト情報の緊急度に応じて、前記ユーザに通知する連絡方
法を変更することを特徴とする請求項1記載の電子秘書
システム。

【請求項5】 前記通知手段は、前記スケジュール情報
に対応する判定ルールを用いて、前記検出されたイベン
ト情報の重要度を計算し、該重要度が大きいイベント情
報ほど前記緊急度が高いと解釈することを特徴とする請
求項4記載の電子秘書システム。

【請求項6】 前記通知手段は、前記重要度と連絡方法
の対応表を備え、該対応表から該重要度に応じた連絡方
法を認識することを特徴とする請求項5記載の電子秘書
システム。

【請求項7】 前記通知手段は、エージェントプロセス
を用いて、前記スケジュール情報に含まれる資源に関す
るイベント情報を前記ユーザに通知することを特徴とす
る請求項1記載の電子秘書システム。

【請求項8】 前記エージェントプロセスは、前記ユー
ザの利用する資源毎に生成され、不要になれば自律的に
消滅することを特徴とする請求項7記載の電子秘書シ
ステム。

【請求項9】 スケジュール管理を行う情報処理システ
ムにおいて、
ユーザのスケジュール情報を管理する管理手段と、
前記ユーザの利用する交通機関に関する障害情報の入力
を検出する検出手段と、
前記スケジュール情報と前記検出手段により検出された

前記障害情報とに基づいて、前記スケジュール情報に含
まれる交通機関に関する障害情報を前記ユーザに通知す
る通知手段と、
を備えることを特徴とする電子秘書システム。

【請求項10】 前記管理手段は、前記スケジュール情
報を路線毎に管理し、前記通知手段は、前記検出された
障害情報を路線毎に該スケジュール情報と比較して、該
検出されたイベント情報が前記スケジュール情報に含ま
れる路線に関係するかどうかを判定することを特徴とす
る請求項9記載の電子秘書システム。

【請求項11】 スケジュール管理を行う情報処理シス
テムにおいて、
ユーザのスケジュール情報および住所情報を管理する管
理手段と、
前記ユーザの利用する資源に関するイベント情報の入力
を検出する検出手段と、
前記スケジュール情報と住所情報と前記検出手段により
検出された前記イベント情報とを用いて、該イベント情
報を前記ユーザに通知する通知手段と、
を備えることを特徴とする電子秘書システム。

【請求項12】 スケジュール管理を行う情報処理装置
により用いられる記憶媒体であって、
該情報処理装置が、
ユーザのスケジュール情報を管理し、
前記ユーザの利用する資源に関するイベント情報の入力
を検出し、
前記スケジュール情報と前記検出手段により検出された
前記イベント情報とに基づいて、前記スケジュール情報
に含まれる資源に関するイベント情報を前記ユーザに通
知するように導くことを特徴とする記憶媒体。

【請求項13】 情報処理装置を用いてスケジュールを
管理する方法において、
ユーザのスケジュール情報を管理し、
前記ユーザの利用する資源に関するイベント情報の入力
を検出し、
前記スケジュール情報と前記検出手段により検出された
前記イベント情報とに基づいて、前記スケジュール情報
に含まれる資源に関するイベント情報を前記ユーザに通
知することを特徴とするイベント通知方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、スケジュールを管
理し、障害が発生した場合にそれをスケジュールに従っ
て行動している人に通知する電子秘書システムおよびそ
の方法に関する。

【0002】

【従来の技術とその問題点】現代社会において生活する
人間は、大抵の場合、各自のスケジュールに従って行動
する。このスケジュールには、少なくとも、何かアクシ
ョンを起こす場合に出発点から終点までどのように移動

すればよいかという情報が含まれている。

【0003】例えば、会社員の場合は、このような移動経路に関する情報は秘書により管理されている場合が多い。したがって、スケジュールに含まれる鉄道路線に事故等の障害が生じている場合、現状では秘書からの連絡がない限り、それを事前に知ることは困難である。また、秘書も常に交通機関の事故発生を監視しているわけではなく、実際にはスケジュールの遂行後に事故発生を知ることが多い。

【0004】このように、スケジュール中に障害が発生している場合でも、それを知るための手段が乏しいため、実際に障害に出会った後に移動経路の変更等の対策を講じている。このため、人の行動が障害発生の後手に回ってしまっているという問題がある。

【0005】もし、スケジュール中の障害等の情報を先に得ることができれば、上述のように対策が後手に回るようなことが起こりにくくなる。したがって、障害が発生していても、その状況の下で、先手を打って最善の行動をとることが可能となる。

【0006】本発明は、個人または団体のスケジュールを妨げる障害情報を、予定時刻以前に通知することができる電子秘書システムおよびその方法を提供することを目的とする。

【0007】

【課題を解決するための手段】図1は、本発明の電子秘書システムの原理図である。図1の電子秘書システムは、管理手段1、検出手段2、および通知手段3を備える。

【0008】管理手段1は、ユーザのスケジュール情報を管理する。検出手段2は、上記ユーザの利用する資源に関するイベント情報の入力を検出する。

【0009】通知手段3は、上記スケジュール情報と検出手段2により検出された上記イベント情報とに基づいて、そのスケジュール情報に含まれる資源に関するイベント情報を上記ユーザに通知する。

【0010】管理手段1は、例えばユーザが移動する際に利用する交通機関を資源とみなし、その利用予定をスケジュール情報として管理する。例えば、このスケジュール情報には、利用する交通機関の路線名や利用日時等のデータが含まれる。

【0011】検出手段2は、外部から入力される交通情報などのイベント情報を検出して、それを通知手段3に伝える。イベント情報としては、大きく分けて、交通事故などの障害が発生したことを示す情報とそれが回復したことを示す情報の2種類がある。

【0012】通知手段3は、スケジュール情報に含まれる資源名とイベント情報に含まれる資源名を比較し、それらが一致した場合にイベント情報をユーザに通知する。これにより、検出されたイベントがユーザのスケジュールに影響を及ぼすかどうか自動的に判定され、影

響のあるイベントの情報だけがユーザに通知される。

【0013】このように、利用予定のある資源に発生した障害等のイベント情報が、検出後直ちにユーザに通知されるので、ユーザは障害に遭遇する前にスケジュールを変更することが可能になる。

【0014】例えば、図1の管理手段1は、実施形態の図9における秘書装置62およびスケジュール管理機構63に対応し、検出手段2はイベント抽出機構64に対応し、通知手段3はイベントブレイス機構65および連絡機構66に対応する。

【0015】

【発明の実施の形態】以下、図面を参照しながら、本発明の実施の形態を詳細に説明する。以下の実施形態においては、イベントの発生場所の解釈を簡単にするために、スケジュール管理の対象となる資源の場所からイベントの発生場所が○(1)の探索時間で分かるような機構を用意しておく。ここで、○(a)は、値aのオーダーの数量を表す。言い換えれば、スケジュールで利用される交通機関等の各資源毎に、その状態を監視するプロセス等の監視機構を用意する。

【0016】図2は、スケジュールを一括して管理するスケジュール管理システムの構成図である。図2のスケジュール管理システムは、管理機構11、スケジュールデータベース(スケジュールDB)12、判定ルールDB13を備える。

【0017】スケジュールDB12には、スケジュールの各要素毎の使用交通機関や路線名、使用開始時刻、終了時刻等を含む複数のスケジュールデータが、一括して格納されている。また、判定ルールDB13には、発生した障害等のイベントがスケジュールに及ぼす影響の大きさを判定するための複数の判定ルールが、各スケジュールデータ毎に一括して格納されている。

【0018】管理機構11は、例えば計算機を含む情報処理装置であり、各スケジュールデータを管理するプロセスを生成して、障害発生時に必要な処理を行う。各プロセスは、例えばある路線上で事故が発生したという情報を、外出先のユーザがその路線を利用する前に携帯電話等に通知する。

【0019】しかし、このような一括管理システムでは、イベントが発生する毎にスケジュールデータとその判定ルールをそれぞれ探索しなければならない。このため、スケジュールデータがn件、それぞれのスケジュールに対するルールがm件ある場合には、最悪O(nm)の探索時間を必要とすることになる。そこで、探索の手間を削減するために、スケジュールデータを分散させることを考える。

【0020】図3は、スケジュールをプロセス毎に分散させて管理する第1の分散管理システムの構成図である。図3の分散管理システムは、管理機構21と判定ルールDB22を備える。管理機構21は、例えば計算機

を含む情報処理装置であり、各プロセスは図2のプロセスと同様の通知処理を行う。

【0021】この管理システムにおいては、管理機構21上の各プロセスにスケジュールデータを分散させ、イベント発生時には、対応するスケジュールデータを持つプロセスが判定ルールDB22から判定ルールを取得する。

【0022】しかし、このような分散管理システムにおいても、やはり各スケジュールデータに対するm件の判定ルールを探索する必要があるため、探索時間はmに依存してO(m)となる。

【0023】図4は、さらに判定ルールも分散させて管理する第2の分散管理システムの構成図である。図4の分散管理システムにおいて、管理機構31は、例えば計算機を含む情報処理装置であり、各プロセスは図2のプロセスと同様の通知処理を行う。

【0024】管理機構31上の各プロセスは、スケジュールデータおよび判定ルールを組にして持つ。したがって、スケジュールデータに対応するイベントが発生した時にデータベースを探索する必要がなく、その判定ルールを用いて直ちにイベントの重要性を判定することができる。また、スケジュールデータの一部を入力パラメータとする評価関数で判定ルールを表すことにより、さらに処理が簡単化される。

【0025】しかし、イベント発生時には、通知等の次の動作を発生させる機構が必要になるため、実施の際に管理機構とは別の機構を加えなければならない。そこで、図4の分散管理システムに、リモートプログラミングに基づく言語であるテレスクリプト (Telescript) による方法を適用することを考える。

【0026】テレスクリプトによるリモートプログラミングにおいては、データを保持するプロセスの一種であるエージェント (Agent) が用いられる。この場合には、スケジュールデータとそれに対する判定ルールを各エージェントに分散させて持たせることが可能であるため、分散管理の形態を取ることができる。

【0027】さらに、エージェントは自律的に機能するという特徴があるため、スケジュールに対応するイベントが発生した場合にも、次の処理を自主的に行うことができる。したがって、次の動作を発生させる別の機構を必ずしも必要としない。

【0028】このテレスクリプトによるリモートプログラミングについては、「リモートプログラミングの実施方法」(特願平6-179767、特開平7-182174)に詳述されているが、ここでその手法について説明する。

【0029】このリモートプログラミングは、複数の計算機システムを結ぶ通信ネットワーク上での処理を記述する方法の一種で、ネットワークを渡り歩く移動可能なプロセスであるエージェントと、エージェントが入って

くる固定プロセスであるプレイス (Place) により実現される。

【0030】図5は、プレイスの構成を示している。プレイスは、他のプロセスと識別可能なテレネーム (tele name) を持ち、任意有限個の子プロセスを持つことができる。また、プレイスは、自己および他のプロセスより参照可能な任意有限個のデータと、自己および他のプロセスより呼出し可能な任意有限個の手続きとを有する。

【0031】さらに、メインルーチンとして手続きliveを持ち、作成されたプレイスはliveを実行する。プレイスは、liveが実行されている間だけ存在し、liveが終了すると消滅する。プレイスが消滅すると、その子プロセスも終了し、消滅する。

【0032】図6は、エージェントの構成を示している。エージェントは、他のプロセスと識別可能なテレネームを持ち、あるプレイスの中の子プロセスとしてのみ存在する。また、エージェントは、自己および他のプロセスより参照可能な任意有限個のデータと、自己および他のプロセスより呼出し可能な任意有限個の手続きとを有する。

【0033】さらに、メインルーチンとして手続きliveを持ち、作成されたエージェントはliveを実行する。エージェントは、liveが実行されている間だけ存在し、liveが終了すると消滅する。エージェントの手続きliveの中でコマンドgoが実行されると、そのエージェントはネットワークを介して指定された行き先へ移動する。

【0034】図7は、コマンドgoによるオペレーションの例を示している。図7において、プレイス41内のエージェント42は、goの実行により、行き先のプレイス44を指定するデータであるチケット (Ticket) 43を得て、指定されたプレイス44に移動する。

【0035】このとき、エージェント42は、goによりliveの実行を一時中断し、保持したデータとともに凍結 (圧縮) され、パック化される。そして、チケット43に示されたプレイス44に送られ、プレイス44により受け入れをチェックされる。プレイス44に受け入れられると、その中に子プロセスとして置かれた後、解凍 (伸張) され、goの次のコマンドからliveの実行を再開する。

【0036】また、エージェントは、手続きliveの中で自己の置かれているプレイス内にある他のエージェントを求めるコマンドmeetを呼出すことができる。コマンドmeetの実行時には、エージェントは相手のエージェントを指定するデータであるペティション (Petition) を用いる。

【0037】図8は、コマンドmeetによるオペレーションの例を示している。図8において、プレイス51内のエージェント52は、コマンドmeetによりあるペティションを指定する。プレイス51は、ペティショ

ンにより指定された条件に該当するエージェントを探す。そして、プレイス51内に該当するエージェントがなければ、そのようなエージェントが入ってくるまで待つ。

【0038】プレイス51内のエージェント53がペティションの条件に該当すれば、プレイス51は、エージェント52が相互作用を希望していることをエージェント53に伝える。そして、エージェント53がエージェント52の希望を受け入れる旨の戻り値を返すと、プレイス51は、エージェント53へのポインタをmeetの戻り値としてエージェント52に返す。その後、エージェント52は、エージェント53へ直接アクセスして、情報交換等を行うことが可能になる。

【0039】以下では、このようなりモートプログラミングの手法を用いてスケジュールの分散管理を行う実施形態について説明する。本実施例では、スケジュール中に発生する障害を先手を打って知るために、スケジュールデータと1対1に対応し、そのスケジュールデータに関する障害等を自律的に判断するエージェントが用いられる。自律性を持つプロセスであるエージェントを用いることにより、スケジュール中の個々の要素を独立に自動的に管理することができる。

【0040】このエージェントは、緊急の度合いを考慮して連絡手段を自動的に判別し、スケジュールに従って行動するユーザに障害等の内容を伝える。これにより、障害が発生していても、ユーザはそのような事象の下で最善の行動を取ることが可能になる。

【0041】また、本実施形態では、事象の重要度を判定する判定ルールをスケジュールデータ毎に関数の形で持たせるため、スケジュールデータに対応した判定ルールを探索する手間が省ける。また、判定ルールを関数化することで、その変更が簡単になり、判定ルールを容易に拡張することが可能になる。

【0042】図9は、本実施形態の電子秘書システムの構成図である。図9の電子秘書システムは、秘書装置62、スケジュール管理機構63、イベント抽出機構64、イベントプレイス機構65、連絡機構66、スケジュールDB61、および住所録DB67を備え、スケジュールを管理する秘書の役割を果たす。

【0043】例えば、秘書装置62、スケジュール管理機構63、イベント抽出機構64、イベントプレイス機構65、および連絡機構66は、それぞれ計算機を有する情報処理装置により実現され、通信ネットワークにより互いに接続されている。

【0044】図10は、これらの装置または機構の一例である情報処理装置の構成図である。図10の情報処理装置は、CPU（中央処理装置）82、メモリ83、通信部84、データベース86、入出力装置87、およびそれらを結ぶバス85を有し、通信部84を介して通信ネットワーク81に接続されている。

【0045】CPU82はメモリ83を利用してエージェントやプレイス等のプロセスを生成し、データベース86のデータを参照しながら必要な処理を行う。生成されたエージェントは通信部84を介して、通信ネットワーク81上の他の情報処理装置内のプレイスに移動することが可能である。入出力装置87は、例えばキーボード等の入力機器を備えたディスプレイ端末である。

【0046】例えば、秘書装置62の場合はデータベース86はスケジュールDB61に相当し、連絡機構66の場合はデータベース86は住所録DB67に相当する。その他の機構については、必ずしもデータベース86を備える必要はない。

【0047】図9において、スケジュール管理機構63はユーザにより操作され、スケジュール情報の入力を受け付けてそれを秘書装置62に伝える。秘書装置62は、受け取ったスケジュール情報に基づいてスケジュール変更処理を行い、スケジュールDB61内の各個人のスケジュールデータを更新する。スケジュールデータは、例えば交通機関の路線のような、ユーザが利用する資源毎に作成される。

【0048】スケジュール変更処理においてスケジュールデータが追加された場合は、秘書装置62は、その利用資源の状態を監視する秘書エージェント71を作成し、イベントプレイス機構65に送り込む。イベントプレイス機構65には、各利用資源に対応するプレイスであるイベントプレイス73が、利用資源の数に応じてあらかじめ複数用意されている。利用資源が移動手段に対応する場合は、例えば、利用する可能性のある全交通機関のすべての路線について、イベントプレイス73が作成される。

【0049】秘書エージェント71は行き先のイベントプレイス73に到着すると、イベントチェック処理を行って、対応するイベントプレイス73の状態を監視する。各イベントプレイス73の状態は、対応する資源に関するイベント情報がイベント抽出機構64に入力された時に更新される。イベント情報は、例えば特定の交通機関に発生した事故等の情報である。イベント抽出機構64はイベント抽出処理によりイベント情報の文字データ等を解析し、解析結果に基づくイベント変更処理を行って、対応するイベントプレイス73の状態を更新する。

【0050】イベントプレイス73の状態は、例えば、対応する資源に障害が発生して利用不可能な状態と、通常の利用可能な状態のうちのいずれかを表す。交通機関の場合は、利用不可能な状態は不通状態である。

【0051】イベントプレイス73の更新により秘書エージェント71はインタラプトを受け、イベントプレイス73が例えば不通または通常という2つの状態のいずれに相当するかを判断することができる。

【0052】また、秘書エージェント71は、秘書装置

62からイベントプレイス機構65に送り込まれた時、すなわちスケジュールDB61が更新された直後に、監視対象のイベントプレイス73が既に不通状態にあれば、そのことを連絡機構66を介してユーザに通知する。イベントプレイス73の状態の更新があっても、例えばそれが不通区間の路線上でさらに別の事故が起こった場合のように、結果的に状態に変化がない時は、秘書エージェント71は何もしない。

【0053】また、秘書エージェント71は、状態に実質的な変化があった時に、その事象の入力時刻と資源を利用する予定時刻との差により事象の重要度を判定する判定ルールを持っている。この判定ルールは秘書エージェント71の作成時に渡されるので、秘書装置62において判定ルールを動的に変更することも可能である。

【0054】イベントプレイス73の状態変化と判定ルールは、秘書エージェント71が作成する連絡エージェント72により連絡機構66に伝えられる。秘書エージェント71は、スケジュールデータに記された資源の利用終了時刻を過ぎると、自発的に消滅する。

【0055】連絡機構66は、連絡エージェント72から状態変化と判定ルールを受け取ると、住所録DB67にあらかじめ格納されている連絡手段データに記述されたユーザの連絡先に、対応する資源の状態を連絡する。例えば、このとき使用される連絡手段は、現在時刻から利用開始時刻までの時間的余裕から判定ルールを用いて算出された重要度に基づいて、切り換えられる。

【0056】次に、図11から図15までを参照しながら、図9の電子秘書システムで用いられるデータの構造について説明する。図11は、スケジュールDB61に格納されるスケジュールデータの構造の例を示している。図11のスケジュールデータは、who、from、to、where、by、および判定ルールを要素として有する。whoはスケジュールに従って行動する主体を表し、例えば個人や団体等のユーザがこれに該当する。

【0057】byは利用する資源を表し、移動手段の場合は交通機関の路線名等がこれに該当する。fromとtoはそれぞれ利用開始日時と利用終了日時を表し、whereは行動主体が存在する場所を表す。例えば、byは山手線や中央線等であり、whereは東京や神奈川等であるが、whereは省略することもできる。

【0058】また、判定ルールは、このスケジュールデータに記された資源に障害が発生したり、それが障害から回復したりした場合に、その状態変化がスケジュールに及ぼす影響の重要度を判定するために用いられる。また、その重要度に基づいて行う具体的なオペレーションも記述している。

【0059】図12は、イベント抽出機構64からイベントプレイス機構65に渡されるイベントデータの構造の例を示している。図12のイベントデータは、by、

when、accident、およびflagを要素として有する。byはスケジュールデータの場合と同様である。

【0060】whenは障害や回復等のイベントの発生日時を表すが、代わりに、イベント抽出機構64にイベント情報が入力した日時をwhenとして用いてもよい。accidentは事故や渋滞等のイベント名を表し、flagはイベントの種別を表す。ここでは、イベントが発生した場合にflagの値をONとし、終了した場合にOFFとする。

【0061】図13は、住所録DB67に格納される連絡手段データの構造の例を示している。図13の連絡手段データは、who、複数の連絡手段のアドレス、および重要度/連絡手段対応表を要素として有する。whoはスケジュールデータの場合と同様である。

【0062】連絡手段のアドレスとしては、電子メールアドレス、ページャ呼出番号、携帯電話呼出番号などが記述される。ページャは、例えばポケットベルのことを指す。重要度/連絡手段対応表には、判定ルールにより算出された重要度に応じて、用いられる連絡手段が記載されている。

【0063】図14は、各イベントプレイス73が保持するデータの構造の例を示している。図14のデータは、by、秘書エージェント在中場所、および事象の並びを要素として有する。byはスケジュールデータの場合と同様である。

【0064】秘書エージェント在中場所は、イベントプレイス73内に送り込まれた秘書エージェント71の存在場所を特定する情報で、例えば秘書エージェント71へのポインタ等がこれに該当する。一般に、イベントプレイス73内には、複数の秘書エージェントが存在することができる。事象の並びには、イベント抽出機構64から渡されたイベントデータが格納される。

【0065】図15は、秘書エージェント71が保持するデータの構造の例を示している。図15のデータは、from、to、判定ルール、およびwhoを要素として有する。これらの情報は、秘書エージェント71の作成時にスケジュールデータからコピーされる。

【0066】次に、図16から図21までを参照しながら、図9の電子秘書システムにおいて行われる各処理について説明する。図16は、秘書装置62およびイベントプレイス機構65により行われるスケジュール変更処理のフローチャートである。図16において処理が開始されると、秘書装置62は、まずスケジュール管理機構63からスケジュールデータとコマンドの対または並びを受け取る(ステップS1)。コマンドは、スケジュールデータの“追加”または“削除”のいずれかを表す。

【0067】次に、受け取ったコマンドが“追加”かどうかを判定し(ステップS2)、“追加”であればスケジュールデータをスケジュールDB61に追加する(ス

10

20

30

40

50

テップS3)。そして、スケジュールデータ中のbyにより行き先を指定したコマンドgoを持つ秘書エージェント71を作成する(ステップS4)。このとき、同時にスケジュールデータ中のwho、from、to、および判定ルールを秘書エージェント71に与えておく。

【0068】作成された秘書エージェント71は、goにより凍結され、イベントプレイス機構65内の指定されたbyを持つイベントプレイス73に移動する(ステップS5)。このとき、該当するイベントプレイス73があるかどうか調べられ(ステップS6)、それが存在すれば、秘書エージェント71はそこでイベントプレイス機構65により解凍される(ステップS7)。解凍された秘書エージェント71は、次にイベントチェック処理を行って(ステップS8)、処理を終了する。

【0069】ステップS6において、該当するイベントプレイス73が存在しなければ、例外処理を行って(ステップS9)、処理を終了する。例外処理においては、例えば、エラーメッセージがスケジュール管理機構63のディスプレイに表示される。この場合、秘書エージェント71はイベントプレイス73には送られない。例外処理は、該当するイベントプレイス73がイベントプレイス機構65に用意されていなかった場合に行われる。

【0070】また、ステップS2においてコマンドが“削除”であれば、秘書装置62は受け取ったスケジュールデータと一致するものをスケジュールDB61内で探してそれを削除し、イベントプレイス機構65にそのスケジュールデータを送る(ステップS10)。

【0071】イベントプレイス機構65は、送られたスケジュールデータのbyに一致するbyを持つイベントプレイス73を探し(ステップS11)、その秘書エージェント71を参照する。そして、スケジュールデータのwho、from、toと一致するデータを持つ秘書エージェント71を探す(ステップS12)。

【0072】該当する秘書エージェント71があるかどうか調べ(ステップS13)、それが存在すればその秘書エージェント71を殺して(終了させて)(ステップS14)、処理を終了する。終了した秘書エージェント71は、自動的に削除される。該当する秘書エージェント71が存在しなければ、そのまま処理を終了する。これは、秘書エージェント71が既に自発的に消滅している場合に相当する。

【0073】図17は、イベント抽出機構64により行われるイベント抽出処理のフローチャートである。図17において処理が開始されると、イベント抽出機構64はイベント入力待ちのループ処理を開始し(ステップS21)、イベント情報が入力されたかどうかを判定する(ステップS22)。イベント情報が入力されなければ、ステップS22の判定を繰り返す。

【0074】交通情報の文字データなどのイベント情報が外部から入力されると、それを解析して図12のよう

なイベントデータを作成し(ステップS23)、イベントプレイス機構65に送付する(ステップS24)。イベント情報は通信ネットワーク81から入力する場合もあり、入出力装置87からオペレータにより入力される場合もある。

【0075】作成されたイベントデータのflagは、新たなイベントの発生時にはONに設定され、既に発生していたイベントの終了時にはOFFに設定される。例えば、事故の発生時にはflagはONに設定され、その回復時にはOFFに設定される。したがって、1つのaccidentに対しては、flag=ONのイベントデータとflag=OFFのイベントデータの2つが作成されることになり、最終的には必ず対になる。

【0076】次に、ループ処理を終了するかどうかを判定し(ステップS25)、終了しない場合はステップS22以降の処理を繰り返す。実際には、このループ処理はイベント入力待ちの無限ループとして設定され、通常は終了しない。ただし、システム自体に異常をきたしたような場合には、処理を終了する。

【0077】図18は、イベントデータを受け取ったイベントプレイス機構65が行うイベント変更処理のフローチャートである。図18において処理が開始されると、イベントプレイス機構65は、まず受け取ったイベントデータのbyに一致するbyを持つイベントプレイス73を探す(ステップS31、S32)。該当するイベントプレイス73があれば、次に、イベントデータのflagがONかどうかを調べる(ステップS33)。

【0078】そして、flagがONであれば、そのイベントプレイス73の事象の並びにイベントデータを追加する(ステップS34)。次に、イベントプレイス73の秘書エージェント71に、そのイベントデータを引数とする割り込みを送り(ステップS35)、処理を終了する。

【0079】ステップS33においてflagがOFFであれば、そのイベントプレイス73の事象の並びからイベントデータのaccidentと一致するaccidentを持つものを探し、それを削除して(ステップS36)、ステップS35の処理を行う。

【0080】また、ステップS32において該当するイベントプレイス73がなければ、例外処理を行って(ステップS37)、処理を終了する。例外処理においては、例えば、エラーメッセージがイベント抽出機構64のディスプレイに表示される。

【0081】図18のイベント変更処理により、イベントが発生した時にイベントプレイス73に事象が追加され、終了すると削除されるので、イベントプレイス73内の事象の並びをチェックすることにより、対応する資源に障害が発生しているかどうかを知ることができる。

【0082】図19は、イベントプレイス73において解凍された秘書エージェント71が行うイベントチェッ

ク処理のフローチャートである。秘書エージェント71は、解凍されると同時に起動される。図19において処理が開始されると、秘書エージェント71は、まず自己が存在するイベントプレイス73の事象の並びに事象（イベントデータ）があるかどうかを調べる（ステップS41）。

【0083】イベントデータがあれば、そのbyと、文字列“事故発生中”と、（from-現在時刻）の値と、判定ルールと、whoとをデータとして保持する連絡エージェント72を作成し、連絡機構66に送付する（ステップS42）。連絡エージェント72は、例えばコマンドsendを用いて生成され、送付される。

【0084】sendは、エージェントの手続きliveの中で実行され、そのエージェントの複製を作って他のプレイスに送るコマンドである。ここでは、連絡エージェント72は、秘書エージェント71のクローンエージェントとして生成される。

【0085】連絡エージェント72が保持するデータの中で、from、判定ルール、およびwhoは秘書エージェント71から複写されて用いられる。また、現在時刻としては、イベントプレイス機構65内のタイマ機構（不図示）から得た日時を用いる。ただし、イベントデータのwhenの値と現在時刻とがあまり変わらない場合には、現在時刻の代わりにwhenを用いてもよい。

【0086】連絡エージェント72を連絡機構66に送ることにより、whoに記されたユーザに“事故発生中”という文字列情報が自動的に伝達される。ステップS41でイベントプレイス73にイベントデータがなければ、そのままステップS43以降の処理を行う。

【0087】次に、秘書エージェント71はループ処理を開始し（ステップS43）、現在時刻が自己の保持するtoを過ぎているかどうかをチェックする（ステップS44）。それがtoを過ぎていなければ、次にイベントプレイス機構65からの割り込みがあるかどうかをチェックする（ステップS45）。

【0088】割り込みがあれば、イベントプレイス機構65から渡されたイベントデータを用いて割り込み処理を行い、イベントの内容を連絡機構66に通知して（ステップS46）、ステップS44以降の処理を繰り返す。割り込みがなければそのままステップS44以降の処理を繰り返す。

【0089】ステップS44、S45、S46のループ処理は、一定時間毎に繰り返される。そして、ステップS44において現在時刻がtoを過ぎれば、もはや該当する資源の状態を監視する必要はなくなるので、自律的に処理を終了して（ステップS47）、消滅する。このように、秘書エージェント71は不要になれば自主的に消滅するので、システム内の計算機資源が効率的に運用される。

【0090】図20は、図19のステップS46におい

て行われる割り込み処理のフローチャートである。図21において処理が開始されると、秘書エージェント71は、まず自己が存在するイベントプレイス73の事象の並びに事象（イベントデータ）があるかどうかを調べる（ステップS51）。

【0091】イベントデータがあれば、ステップS42と同様にして、byと、文字列“事故発生中”と、（from-現在時刻）の値と、判定ルールと、whoとをデータとして保持する連絡エージェント72を作成して、連絡機構66に送付し（ステップS53）、処理を終了する。

【0092】また、イベントデータがなければ、引数として渡されたイベントデータのbyと、文字列“回復”と、（from-現在時刻）の値と、判定ルールと、whoとをデータとして保持する連絡エージェント72を作成して、連絡機構66に送付し（ステップS52）、処理を終了する。

【0093】こうして、イベントプレイス73内の事象の並びが空かどうかに応じて、文字列“回復”または“事故発生中”が連絡機構66に伝えられる。尚、イベントプレイス73の事象の並びにイベントデータが複数ある場合は、文字列を“多重事故発生中”としてもよい。また、イベントデータの個数が多くなるほど重要度が大きくなるようなオペレーションを、判定ルールとして用意しておいてもよい。

【0094】図21は、連絡エージェント72を受け入れた連絡機構66が行う連絡処理のフローチャートである。図21において処理が開始されると、連絡機構66は、まず連絡エージェント72からby、文字列、（from-現在時刻）の値、判定ルール、およびwhoを受け取る（ステップS61）。連絡エージェント72は、連絡機構66にこれらのデータを渡すと自律的に消滅する。

【0095】次に、連絡機構66は（from-現在時刻）をパラメータTとする判定ルールを用いて、そのイベントの重要度を計算する（ステップS62）。重要度は、Tが小さいほど、すなわち現在時刻から開始時刻までの時間的余裕が少ないほど大きくなるように定義する。

【0096】ここでは、例えば、Tの逆数または T^2 の逆数を重要度を表す関数として用いる。そのほかに、Tが一日以内なら重要度を大、Tが一週間以内なら重要度を中、Tが一週間を超えるなら重要度を小とするような関数を用いてもよい。

【0097】次に、住所録DB67からwhoに対応する図13のような連絡手段データを取得し（ステップS63）、その重要度／連絡手段対応表を参照する（ステップS64）。そして、求めた重要度の値に対応する連絡手段を選択し、そのアドレスを連絡手段データから取得する。次に、そのアドレスを用いて、選択した連絡手

段でユーザに連絡して(ステップS65)、処理を終了する。

【0098】例えば、whoに記されたユーザが携帯電話、ページャ、および電子メールを連絡手段として登録している場合は、重要度に適当な閾値を設けて、重要度が大きければ携帯電話、中くらいならばページャ、小さければ電子メールが選択されるようにしておく。携帯電話には、例えば音声合成で生成された音声により自動的に連絡される。こうしておけば、緊急度の高いイベントの変化ほど迅速な連絡手段でユーザに通知されることになる。

【0099】連絡する内容は、連絡エージェント72から受け取ったbyと文字列を用いて、例えば“山手線に事故発生中”または“山手線が回復”のように構成される。連絡を受けたユーザは、その情報に基づいてスケジュールの変更をスケジュール管理機構63に指示するなどの適当な対策を講じることができる。

【0100】以上説明したリモートプログラミングによる電子秘書システムにおいて、秘書装置62、スケジュール管理機構63、イベント抽出機構64、イベントブレイス機構65、および連絡機構66のうちのいくつかまたは全部を、同一の情報処理装置上で実現してもよい。また、イベントブレイス機構65から連絡機構66へは、連絡エージェント72によりデータが伝えられているが、その代わりに秘書エージェント71がデータのみを直接送付するようにしてもよい。

【0101】また、スケジュールリングされる資源の例として交通機関を用いて説明したが、本発明はこれに限らず、例えば会議室、備品、設備等の他の資源についても同様に適用できる。この場合、対応する資源の識別情報がスケジュールデータ等のbyとして用いられる。そして、その資源が何等かの理由で利用不可能となった時に、イベントデータのflagがONに設定され、再び利用可能となった時にそれがOFFに設定される。

【0102】さらに、図9の実施形態において、必ずしもエージェントやブレイスを用いる必要はなく、それぞれの処理を普通のプロセスにより行ってもよい。この場合は、ネットワークを介したやりとりは、例えばRPC(remote procedure call)により実行される。

【0103】

【発明の効果】本発明によれば、スケジュールされた利用資源に発生した障害等のイベント情報を、その利用予定日時より前にユーザに通知することが可能になる。これにより、連絡を受けたユーザは障害の発生した資源の利用予定をキャンセルして、代わりに他のスケジュールを優先させるなどの処置を取ることができる。したがって、障害に遭遇する前に先手を打って、最善の行動をとることが可能となる。

【図面の簡単な説明】

【図1】本発明の原理図である。

【図2】一括管理システムの構成図である。

【図3】第1の分散管理システムの構成図である。

【図4】第2の分散管理システムの構成図である。

【図5】ブレイスの構成を示す図である。

【図6】エージェントの構成を示す図である。

【図7】オペレーションgoを示す図である。

【図8】オペレーションmeetを示す図である。

【図9】電子秘書システムの構成図である。

【図10】情報処理装置の構成図である。

10 【図11】スケジュールデータの構造を示す図である。

【図12】イベントデータの構造を示す図である。

【図13】連絡手段データの構造を示す図である。

【図14】イベントブレイスの持つデータの構造を示す図である。

【図15】秘書エージェントの持つデータの構造を示す図である。

【図16】スケジュール変更処理のフローチャートである。

【図17】イベント抽出処理のフローチャートである。

20 【図18】イベント変更処理のフローチャートである。

【図19】イベントチェック処理のフローチャートである。

【図20】割り込み処理のフローチャートである。

【図21】連絡処理のフローチャートである。

【符号の説明】

1 管理手段

2 検出手段

3 通知手段

11, 21, 31 管理機構

30 12, 61 スケジュールデータベース

13, 22 判定ルールデータベース

41, 44, 51 ブレイス

42, 52, 53 エージェント

43 チケット

62 秘書装置

63 スケジュール管理機構

64 イベント抽出機構

65 イベントブレイス機構

66 連絡機構

40 67 住所録データベース

71 秘書エージェント

72 連絡エージェント

73 イベントブレイス

81 通信ネットワーク

82 CPU

83 メモリ

84 通信部

85 バス

86 データベース

50 87 入出力装置

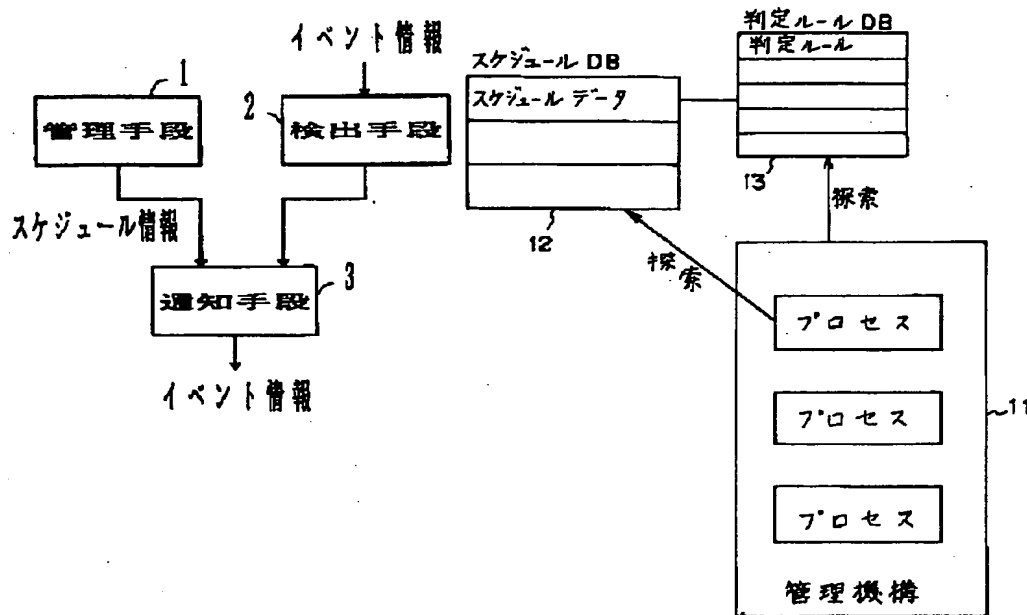
【図1】

【図2】

【図6】

本発明の原理図 一括管理システムの構成図

エージェントの構成を示す図

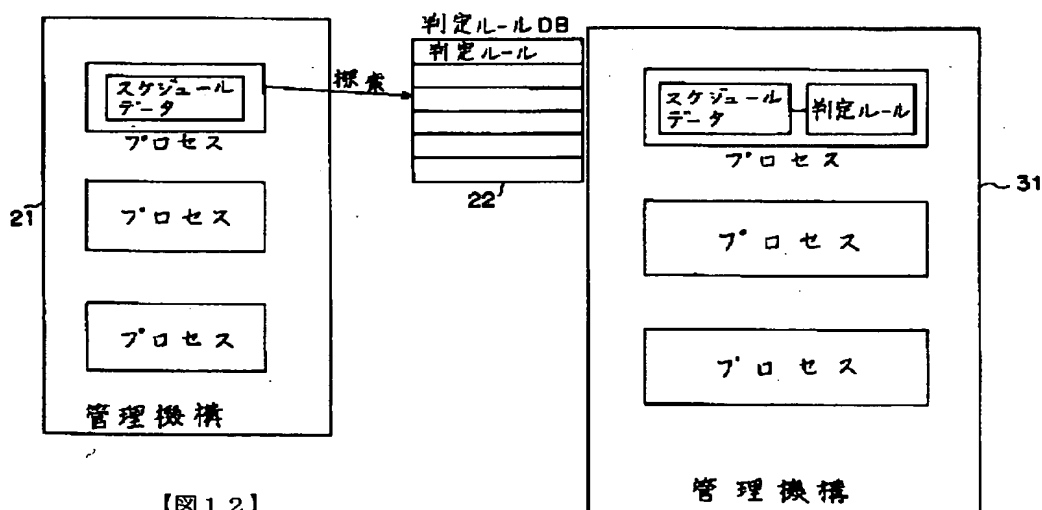


【図3】

【図4】

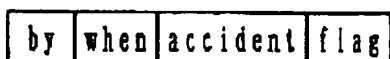
第1の分散管理システムの構成図

第2の分散管理システムの構成図



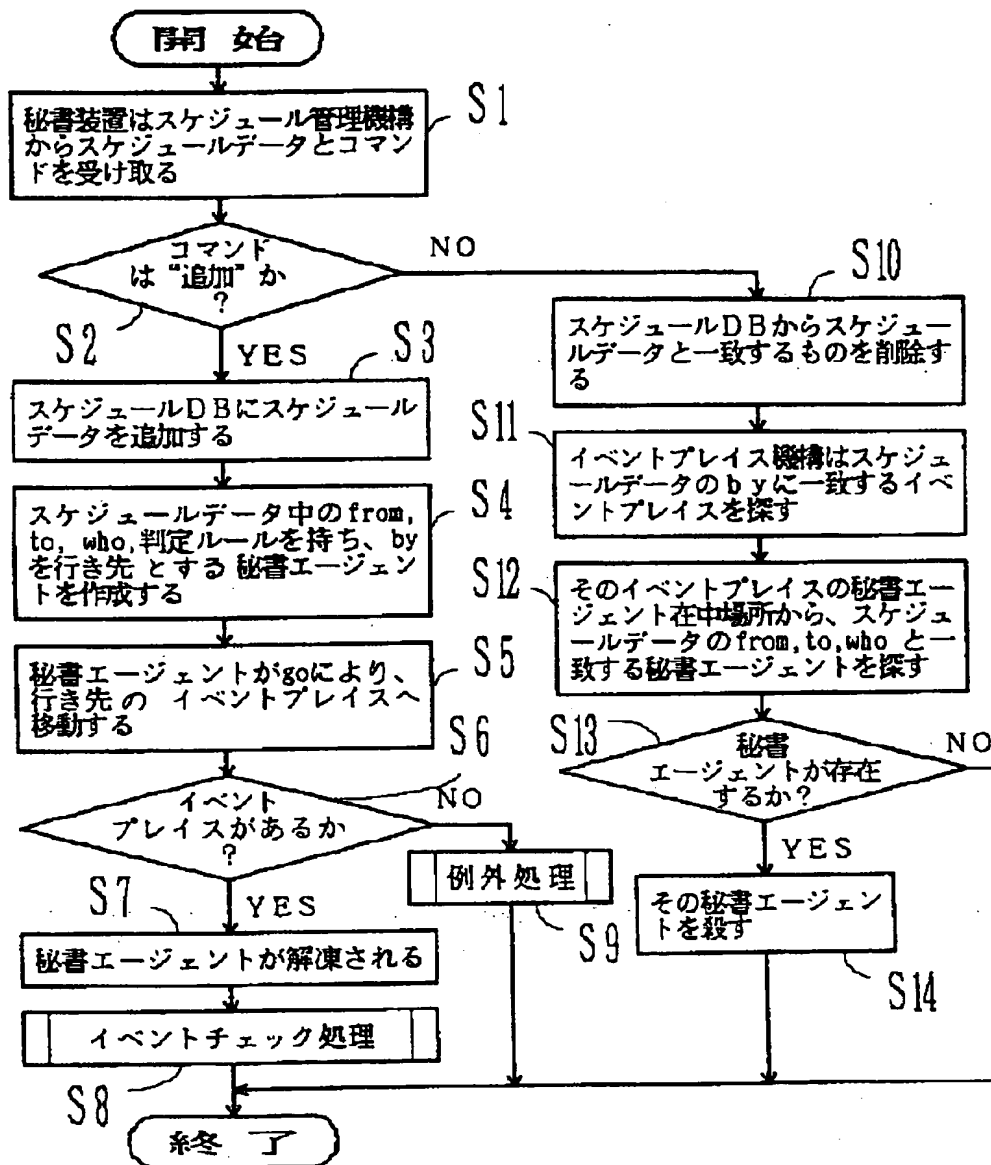
【図12】

イベントデータの構造を示す図



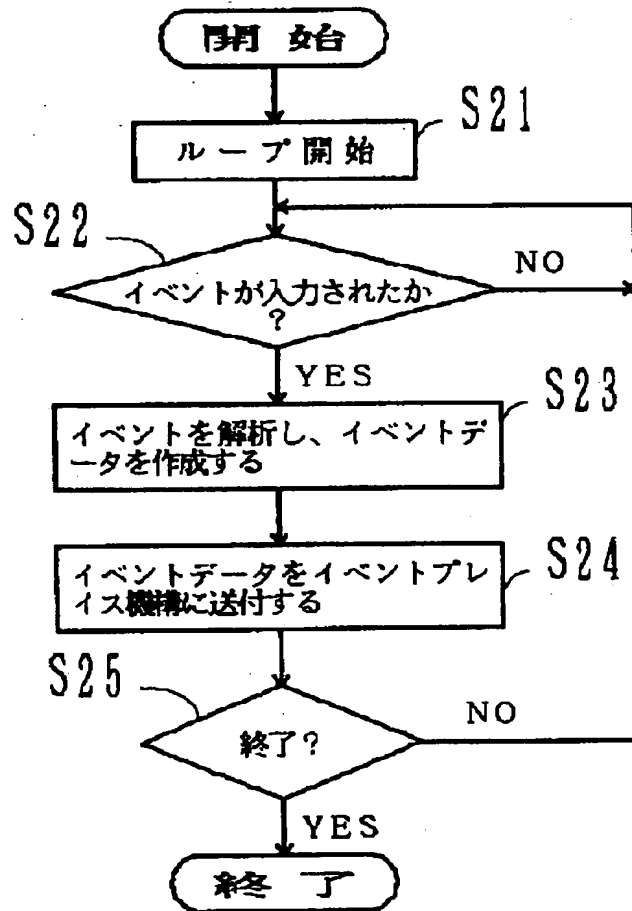
【図16】

スケジュール変更処理のフローチャート



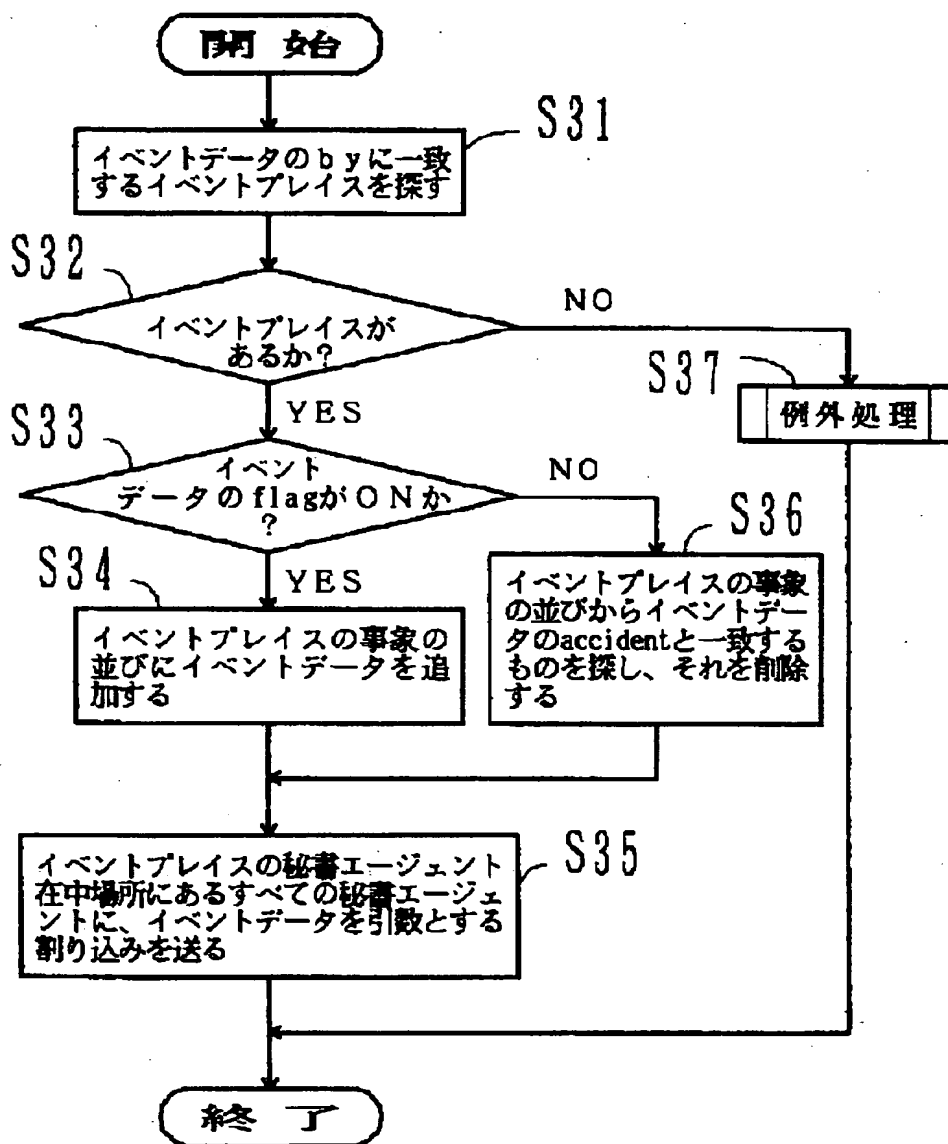
【図17】

イベント抽出処理のフローチャート



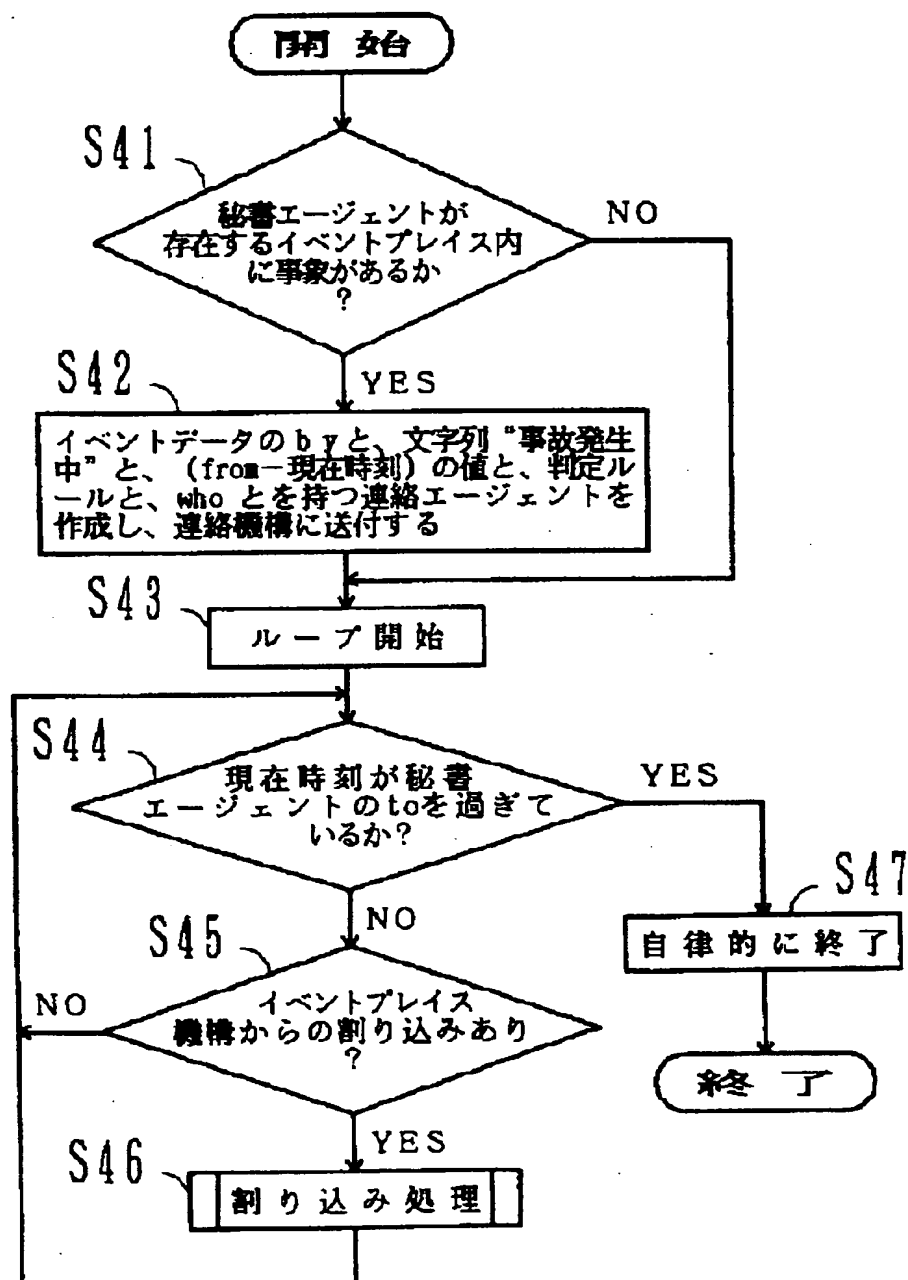
【図18】

イベント変更処理のフローチャート



【図19】

イベントチェック処理のフローチャート



【図20】

割り込み処理のフローチャート

